IBGP confederation provisioning

Mohamed Nassar, Radu State, Olivier Festor

IBGP confederation provisioning

M. Nassar, R. State and O. Festor

LORIA - INRIA Lorraine
615, rue du jardin botanique
54602 Villers-les-Nancy, France
Email: {nassar,state,festor}@loria.fr

Abstract. *This paper proposes an optimization method for the design of large scale confederation based BGP networks. We propose a graph based model and an associated metric to evaluate the reliability of large scale autonomous systems. We propose and validate an effective methodology to find the optimal design for a given physical topology. According to our experiments, we consider that replacing the traditional IBGP topology by an appropriate confederation design could increase at the same time the scalability and the reliability into the domain. Our work might be a step further towards a large scale confederation deployment.*

1 Introduction

The confederation topology is one solution to control IBGP scalability into a large Autonomous System. Although, some general guidelines propose to follow the physical topology and use a hub-and-spoke architecture [9], a dedicated analytical design methodology has not yet been developed. This issue is of extreme importance for large networks and complex topologies. Questions such as "how many sub-AS do we need?" and "where is the border of each sub-AS?", do not have answers based on a theoretical approach.

The paper is organized as follows: Section 2 introduces the BGP protocol and highlights the scalability problem and the current approaches to deal with. Section 3 presents the requirements of confederation reliability and gives hints for optimal confederation design. Section 4 presents a network model and proposes metrics and constraints to create a confederation framework. Solving of the reliability-aware design problem together with implementation and experimental results are in section 4 as well. Section 5 concludes the paper.

2 BGP protocol and scaling large ASs

Today's Internet is structured according to separate administrative domains, called *autonomous systems* ASs, where each has its own independent routing policies. *The Internal Gateway Protocol* IGP is responsible for packets forwarding within a domain. *The Border gateway protocol* BGP is currently the de facto standard protocol for inter domain routing in the Internet. The routers

running BGP are called *speakers*, and a *neighbor connection* (also referred as *peer connection*) can be established between two speakers over TCP. If the two speakers are within the same AS, BGP is called *internal BGP* (IBGP), while two speakers residing in two different ASs and directly attached by a physical segment can established a BGP session and in this case we have an *external BGP* session (EBGP). The speakers using EBGP are called *border routers*.

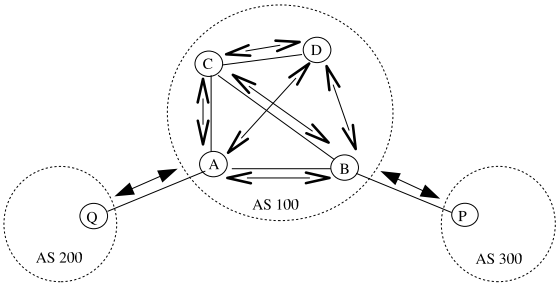


Fig. 1. IBGP and EBGP

Figure 1 shows an example of three ASs, the nodes represent BGP speakers and the solid lines represent physical links. We have two EBGP sessions between A and Q and between B and P, which are border routers, and six IBGP sessions forming a logical full mesh. The border routers A and B inform all the speakers within the domain (by IBGP) about the reachable network addresses outside the domain (learned by EBGP).

EBGP speakers can detect routing loops by the AS-path BGP attribute. But inside the AS, a full mesh of IBGP sessions between speakers is required. The problem with the IBGP mesh is that it is not scalable. If a mesh between n routers has to be deployed, each router handles concurrently $n-1$ sessions. As n grows, routers with higher CPU power and larger memory are required to process and maintain routing information. To solve the IBGP scalability problem, the network community has proposed two practical approaches: Route Reflection and confederation [3].

The route reflection method elects some routers to be route reflectors, and then clusters are formed by assigning clients to each route reflector. The full mesh is only required between reflectors and each client only communicates with its reflector. This method has advantages such as low migration complexity because there is no need to reconfigure all the routers, and it supports hierarchical structures.

The underlying idea of the confederation is to divide a large AS into a number of smaller autonomous systems, called *member AS* or *sub-AS*. Each sub-AS will have a different AS number. The direct consequence is that External BGP sessions must be deployed between them. This sessions are called *intra confederation* EBGP sessions, because they are slightly different of the regular EBGP

sessions. Inside each sub-AS, a full IBGP mesh is required, but we can also deploy a route reflection architecture.

From the outside, a confederation looks like a single AS, because it doesn't expose its internal topology when advertising routes to EBGP neighbors. An exterior AS, basing its routing policy on AS path length, will count a confederation like one hop while the traffic may pass through multiple sub-ASs. Since there may be a shorter path that doesn't include the confederation, this will cause sub-optimal routing. Moreover, in standard BGP, sub-ASs do not alter the overall AS-path length, which causes sub-optimal routing inside the confederation.

The advantage of the confederation design with respect to the route reflectors design is its scaling potential for the IGP protocol. An IGP protocol can be run on one sub-AS totally independent from running other IGPs on other sub-ASs, which helps to control the instability of IGP in a large autonomous system. For more details on BGP, route reflection and confederation issues, the reader is invited to consult the excellent BGP overview in [3].

3 Guidelines for optimizing confederation networks

A good BGP network design must satisfy the following requirements: reduced complexity, simple routing procedure and in the same time high reliability.

The *hub-and-Spoke architecture* is advised in the literature([9],[3]). One sub-AS forms a backbone and play the role of a transit center. All other members connect exclusively with it. The goal of such design is to reduce the number of intra-confederation EBGP sessions, because if a sub-AS has multiple EBGP sessions, it will receive multiple copies of the same routes, which means redundant traffic and processing. The other benefit is the consistency and the predictability of the routing. Uniformly, a traffic entering the confederation from one sub-AS will take two hops to get out by another sub-AS.

But in term of network resilience, a reduced number of intra confederation sessions may be a bad design in case of component failures: for example if one sub-AS is connected to the backbone sub-AS via one session carried by one physical link, the failure of this link or one of the two end routers causes the complete isolation of this sub-AS from the rest of the confederation. A second example is when multiple sub-ASs are connected to the backbone sub-AS and all the sessions are initiated exclusively with the same router. The failure of this transit router transforms the confederation into islands. In backbone networks, there is a small probability that two components fail in the same time, or that the second component fails before we recover the first one. Under this assumption, a topology where there are two independent sessions formed by independent physical components (router, physical-link, router) between every sub-AS and the transit sub-AS, prevents the isolation between sub-ASs.

The authors in [8] propose an IBGP route reflection optimization algorithm, based on the expected session loss metric. This work is focused on optimizing route reflection architectures. The damage caused by a BGP session failure is: 1) the invalidation of routing entries, which are directly or indirectly related to

this session, 2) the consequent route flaps, 3) the unreachable network addresses, or 4) the potential isolation of two parts of the network. Inside each sub-AS, an IBGP mesh must be deployed. When two routers don't have a direct physical link to build a peer session, they use IGP routing tables to make a multi-hop TCP connection and establish an IBGP session. The result is that some physical links will support multiple sessions, and some routers may be also in the path of sessions that it doesn't initiate. When a component (router or link) fails, the overlying sessions may break down.

The session failure is of probabilistic nature [4]. If a router fails, all the initiated sessions will break down, and with certain probability the sessions which pass through it will also fail. If a physical link fails, then each of its overlying sessions may break down with certain probability.

A good sub-AS design should prevent a high expected session loss. The guideline is to follow the logical topology by the physical topology [9]. A sub-AS structure with a physical segment for every two of its IBGP speakers, limits the loss to probably one session per link failure, and certainly all the initiated sessions per router failure.

4 Reliable confederation topology design

4.1 Network models

We represent the physical network in the AS as a graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ represent the set of routers and $\mathcal{E}$ represent the set of physical links. We denote $(i, j) \in \mathcal{E}$ the edge between node $i \in \mathcal{V}$ and node $j \in \mathcal{V}$. Typically, there are some routers that don't run BGP, we denote $\mathcal{V}_r$ the set of routers running BGP, $\mathcal{V}_r \subseteq \mathcal{V}$, and we define $n = |\mathcal{V}_r|$ as the number of BGP speakers. We focus on a transit domain where $\mathcal{V} = \mathcal{V}_r$, and we consider that our model can be simply extended to be applicable on a general case. A reliability model is inherently bounded to the reliability of single components like routers and physical links. The reliability of a router is strongly related to its resource consumption (CPU for route processing, and memory for routing table). When the number of sessions handled concurrently increases over a certain threshold, the router can no longer maintain an up-to-date map. In a confederation topology, except border routers, a router must manage sessions just with the speakers of its sub-AS rather than all the speakers of the AS. The scalability problem is solved this way. Let v_i be the proportion of time where router i has a healthy status. v_i can be assigned based on monitoring history or estimated basing on CPU performance and memory capacity. Likewise, we represent the reliability of a link (i, j) by a value w_{ij}, which is the proportion of time where the link works properly. If no physical link between i and j, $w_{ij} = 0$.

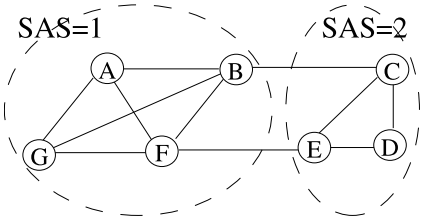
In a logical topology formed by k sub-ASs, each sub-AS is represented by a sub-graph and assigned a number SAS, $1 \leq SAS \leq k$. The logical model $G(\mathcal{V}, \mathcal{E}, f)$ is obtained by characterizing the physical model by a function $f : \mathcal{V} \mapsto [1, k]$. f assigns for each node the sub-graph that contains it. The main

property of f is that it divides the graph into connected sub-graphs. Basing on f, we can calculate the number of nodes of a sub graph by the formula: $y(SAS) = card(\{i \in \mathcal{V}; f(i) = SAS\})$. The number of edges between the nodes of the same sub-graph can be also calculated: $m(SAS) = card(\{(i,j) \in \mathcal{E}; f(i) = f(j) = SAS\})$. We can denote the border routers by a function b: $b(i) = 1$ if $\exists j \in \mathcal{V}; (i,j) \in \mathcal{E} \wedge f(i) \neq f(j)$. So $b(i) = 1$ if i is a border router and 0 else. To build an EBGP session, two border routers must be in different sub-ASs. We use a function s to detect this property, $i, j \in \mathcal{V} : s(i,j) = 1$ if $f(i) \neq f(j)$ and 0 otherwise.

4.2 Problem statement

Given the physical network topology $\mathcal{G}(\mathcal{V}, \mathcal{E})$ of an autonomous system, find among all the possible logical confederation topologies, the one having the best reliability.

For example, we give the physical topology in figure 2. We suppose that one or more of the seven routers don't have the necessary performances to handle six sessions concurrently. The problem is to divide the routers in a number of sub-ASs and to optimize the reliability of the routing protocol.



i	A	G	B	F	D	C	E
$f(i)$	1	1	1	1	2	2	2
$b(i)$	0	0	1	1	0	1	1

Fig. 2. Physical topology and associated solution-logical topology

The solution for this topology is depicted in the table of figure 2. A theoretical justification of this choice can't be completed without a study of the factors that influence either IBGP or EBGP reliabilities. We will model these factors by a suitable metric accompanied by three essential constraints.

4.3 Density metric and accompanying constraints

We define a metric capable to evaluate the difference between the physical topology of a sub-graph and a Clique [2] of the same size. The motivation for our approach is that a Clique has the weakest expected session loss and the highest edge connectivity [1] (for a Clique of size n nodes the edge connectivity is n-1). Our approach is to cut the network into a small number of dense sub-ASs. The density notion was used in [7] to characterize the Internet hierarchy. We define

the Density of a sub-graph as the ratio of the number of its edges m to the number of edges required to accomplish a Clique between its nodes. For n nodes, we need $\frac{n\times(n-1)}{2}$ edges to make a Clique.

$$D(SAS) = \frac{m}{\frac{y(SAS)\times(y(SAS)-1)}{2}}$$

We define the density of a graph k-cut (i.e. the graph is cut into k connected sub-graphs) as the average of densities of its sub-graphs.

$$D = \frac{\sum_{SAS=1}^{k} D(SAS)}{k}$$

A logical topology which concentrates the edges in the sub-graphs reduces in the same time the number of edges between the sub-graphs, and that the number of EBGP sessions is minimized.

To address the EBGP resilience, we introduce here the cut reliability constraint. We define the reliability of intra-confederation EBGP as the sum of reliabilities of the underlying network components (border routers and physical links) and we denote it by R. R indicates approximately how many components deploy EBGP and how much these components are reliable.

$$R = \sum_{i\in\mathcal{V}} v_i \times b(i) + \sum_{(i,j)\in\mathcal{E}} w_{ij} \times b(i) \times b(j) \times s(i,j)$$

Our constraint requires that R should be greater than a certain threshold weighted by a fraction α to the sum of reliabilities of the components of all the network.

$$R_T = \sum_{i\in\mathcal{V}} v_i + \sum_{(i,j)\in\mathcal{E}} w_{ij}$$

The second constraint that we have used is limiting the number of sub-ASs. The intra-confederation EBGP routing is not optimal without manually setting BGP policies. When the number of sub-ASs increases, the IGP advantages become non relevant. Thus, we choose not to exceed a certain threshold of number of sub-ASs, otherwise we need much administration effort to save the stability and the efficiency on the routing plan.

Finally, it is important to uniformly distribute the routers among the sub-ASs. We balance between the numbers of IBGP sessions that a router will handle concurrently, what protects certain routers from unsupportable resource consumption, and we balance between the different IGP's working in the sub-ASs. The third constraint is so called the load balancing constraint.

4.4 Reliable confederation-density (RC-D) problem

Given a graph $\mathcal{G}(\mathcal{V},\mathcal{E})$, $\{v_i\}$ and $\{w_{ij}\}$ reliability values of nodes and edges, the RC-D problem aims to find k and the k-cut of the graph which maximize the density metric while respecting the three constraints formulated below:

1. The cut reliability constraint: $R > \alpha \times R_T$;
2. The number of sub-AS constraint: $2 \leq k < \lceil \ln(n) \rceil$;
3. The load balancing constraint: $\forall SAS; \beta \times \frac{n}{k} < y(SAS) < \frac{(2-\beta)\times n}{k}$ where we choose β from $[0.5, 0.9]$.

We can choose α and β and change the threshold of k to strengthen or relax the constraints. A good choice requires practical experience and studying BGP confederation history examples.

4.5 Heuristic solution for reliable confederation topology design

If k is fixed and the graph will be divided on exactly k sub-graphs, then we get the k-RC-D problem. Solutions of the k-RC-D problem for k going from 2 to $\lceil \ln(n) \rceil$ can be compared to elect the optimum design. In this paper, we apply a technique similar to one of the Min k-cut problem solving methods [6].

Our solution HS fixes k first and uses a randomized procedure called *contract* next to divide the graph into k connected sub-graphs. the Contract procedure chooses an edge from $\mathcal{E}$ randomly (the same probability for all the edges). The chosen edge is erased and its two extremities are joined in one meta node. The edges of each of the two extremities belong now to the new meta node. This contraction is repeated iteratively and stops when we reach k meta nodes. The nodes compacted on each meta node are returned as a connected sub-graph. The output of this procedure is a logical topology and the associated function f is represented by a list that assigns for every node in $\mathcal{V}$ the SAS of the sub-graph containing it.

Next, HS calculates the cut reliability R, and the number of routers for each sub-graph y(SAS). If the topology exceeds the reliability constraint or the load balancing one, HS gives it a null density. Otherwise, HS calculates the density of each sub-graph and then the average density. HS repeats this work (*contract+metric calculation*) for $n^2 \times \log n$ iterations like in the algorithm of the Min k-cut to increase the chance to be close to the optimal solution. At the end of this loop, HS picks the maximum density and the associated list representing f as the response to the k-RC-D problem. To respond to the RC-D problem, HS assigns to k all the integer values between 2 and $\lceil \ln(n) \rceil$, solves each of the k-RC-D problems, and finally returns among all the k-RC-D solutions the one having the maximum density. Thus, the complexity of our solution is $O(n^4(\ln(n))^2)$ because the complexity of *contract* procedure is $O(n^2)$. The pseudo-code of HS is depicted below:

```
for k = 2 to ⌈ln(n)⌉
        for topology= 1 to n^2 × log n
                f[topology]=contract(G);
                if (f[topology] satisfies constraints):
                        D_top[topology] = calculate_D(f[topology])
                else:
                        D_top[topology]=0
```

D_k[k]= max(D_top)
D_opt=max(D_k)
return(D_opt,k_opt,f_opt)

4.6 Experimental results

We have implemented a brute force algorithm (ET) which works in exponential time (k^n), tries all combinations, generates all possible logical topologies and returns exactly the maximum possible density. Our objective is to compare the results of HS and those of ET.

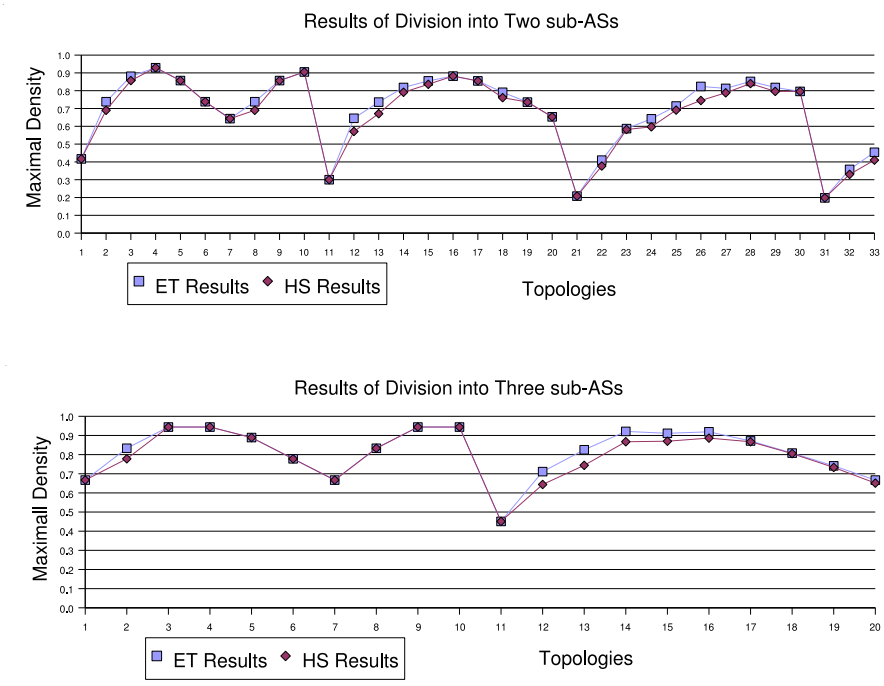


Fig. 3. experimental results(1)

Physical network topologies are generated using the BRITE network topology generator [5]. We have chosen BRITE because it is one of the generators commonly used in the networks and Internet research community. We have chosen to use the Heavy-tailed distribution to place the nodes and the Waxman model to interconnect them. The reliabilities of physical links, w_{ij}, are generated randomly from the interval $[0, 1.9]$ and the reliabilities of routers, v_i, from

$[0, 0.99]$. We choose $\alpha = \frac{1}{n}$ for the cut reliability constraint and $\beta = 0.5$ for the load balancing constraint. We have generated 33 physical topologies: 10 for every network size of 10, 15 and 20 nodes, and 3 for the size of 25 nodes. For each topology, we decided to cut the graph into two sub-graphs, so we fixed k at 2, and we executed the two algorithms. Because it's much harder for ET to cut the graph into 3 sub-graphs for topologies of twenty nodes and more, we did the comparison only for the first twenty topologies of sizes 10 and 15 nodes. The two diagrams in figure 3 show the difference between the two algorithms.

For a given topology, the density of the optimal confederation design returned by the ET algorithm is noted D_{ET} and the density of the one returned by the HS algorithm is noted D_{HS}, thus the relative error for a given topology is:

$$e_r = \frac{D_{ET} - D_{HS}}{D_{ET}} \times 100.$$

For each set of topologies of the same size, we have compared the average relative error and the maximum relative error. The results are shown in the two diagrams of figure 4. After interpreting these diagrams, we have concluded that the HS algorithm could be a good solution to solve the RC-D problem.

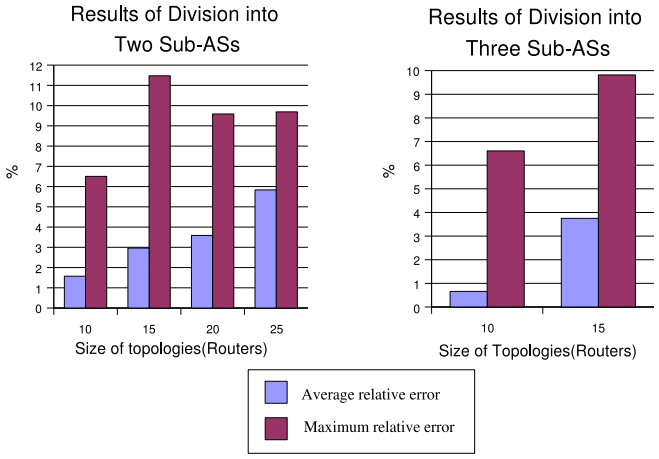


Fig. 4. experimental results(2)

5 Conclusion

We have proposed in this paper a new method for optimizing BGP confederation networks. Our approach consists on determining a criteria for the sub-AS IBGP resilience, as well as its integration in the global EBGP resilience model.

We have adopted a randomized algorithm to optimize the confederation design with respect to our defined resilience, and we have experimentally evaluated its performance.

Acknowledgment This paper was supported in part by the EC IST-EMANICS Network of Excellence (#26854).

References

1. Richard Cole and Ramesh Hariharan. A fast algorithm for computing steiner edge connectivity. In *STOC 03: Proceedings of the thirty-fifth annual ACM symposium on Theory of computing*. ACM Press, 2003.
2. Thomas H. Cormen, Clifford Stein, Ronald L. Rivest, and Charles E. Leiserson. *Introduction to Algorithms*. McGraw-Hill Higher Education, 2001.
3. S. Halabi and McPherson D. *The definitive BGP resource : Internet Routing Architectures Second Edition*. Cisco Press, 2000.
4. L. Xiao L. and K. Nahrstedt. Reliability models and evaluation of internal bgp networks. In *IEEE INFOCOM 2004, Hong Kong, China*, March 2004.
5. A. Medina, A. Lakhina, I. Matta, and J. Byers. Brite: Universal topology generation from a user's perspective (user manual). Technical report, Boston University, 2001.
6. Rajeev Motwani and Prabhakax Raghavan. Randomized algorithms. *SIGACT News*, 26(3), 1995.
7. L. Subramanian, S. Agarwal, J. Rexford, and R. Katz. Characterizing the internet hierarchy from multiple vantage points. Technical report, University of California at Berkeley, 2001.
8. L. Xiao, J. Wang, and K. Nahrstedt. Reliability-aware ibgp route reflection topology design. Technical report, Department of Computer Science University of Illinois at Urbana-Champaign, August 2003.
9. R. Zhang and M. Bartell. *BGP Design and Implementation : Practical guidelines for designing and deploying a scalable BGP routing architecture*. Cisco Press, 2004.